🛠️ Ejercicio Práctico

📌 **Título:** Diagnóstico y Propuesta de Mejora en una Pipeline CI/CD con Falencias de Seguridad

---

🎯 Objetivo:

Evaluar críticamente un flujo de integración continua (CI/CD), identificar debilidades en seguridad y proponer una arquitectura de automatización segura que integre herramientas como **SAST, DAST, SCA y escaneo de infraestructura**.

🧱 Escenario:

La empresa **CodeSecure Ltd.** ha implementado un pipeline básico de CI/CD para su aplicación web. El pipeline actual realiza lo siguiente:

- Compila el código y lo prueba con Jest.
- Despliega automáticamente el sistema a producción.
- No hay validación de seguridad ni control de dependencias.
- No se realiza ningún análisis del entorno (variables, imágenes Docker, configuración de red, etc).
- No hay control de roles ni aprobación manual antes del despliegue.

La organización está preocupada por la posibilidad de introducir código vulnerable en producción y te ha solicitado asesoría.

📋 Actividades:

1. **Detecta al menos 4 debilidades de seguridad en el pipeline descrito.**

2. **Propón una solución técnica concreta para cada una (puedes mencionar herramientas específicas).**

3. **Diseña un mini-diagrama (opcional) o explica el flujo ideal del pipeline con medidas de seguridad automatizadas.**

4. **Menciona al menos 2 ventajas estratégicas de adoptar este pipeline seguro en un entorno DevSecOps.**

📌 Formato sugerido de respuesta:

Debilidad Detectada	Herramienta / Técnica Propuesta	Justificación
Falta de análisis del código fuente		

💡 Herramientas recomendadas (puedes usar otras):

- **SonarQube** (SAST)
- **Snyk** o **OWASP Dependency-Check** (SCA)
- **OWASP ZAP**, **Burp Suite CLI** (DAST)
- **Trivy**, **Checkov**, **Docker Scout** (escaneo de contenedores y configuración)
- **GitHub Actions, GitLab CI, Jenkins**, etc.
